AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions and listings of claims in the application:

LISTING OF CLAIMS:

1 (currently amended): A flash memory having a map block, the map block comprising:

a first mapping table containing a physical address allocated to a block of a plurality of blocks, wherein the plurality of blocks constitute a data block and status information of each of the plurality of blocks;

a second mapping table containing mapping information between the physical address and a ~~local~~ logical address of each of the plurality of blocks in the first mapping table from which error blocks are excluded; and

a third mapping table in which most recent mapping information is written and processed by a specified value to minimize an update operation of the second mapping table.

2 (original): The flash memory as claimed in claim 1, wherein the map block further comprises a first, a second, and a third spare block for the respective first, second, and third mapping tables.

3 (original): The flash memory as claimed in claim 2, wherein the second and third spare blocks store previous mapping information which is updated due to mapping information generated by write and delete operations.

4 (original): A flash memory access apparatus, comprising:

a flash memory including a first mapping table, a second mapping table, a third mapping table and respective first, second, and third spare blocks, wherein the first mapping table contains a physical address information of a data block, the second mapping table contains mapping information of the first mapping table from which error blocks are excluded, and the third mapping table contains most recent mapping information; and

a flash memory controller for generating a fourth mapping table containing free block information through the first, second and third mapping tables obtained from a map block in the flash memory, and for accessing respective physical addresses from and into which data will be read and written by referring to the second and third mapping tables in read operations and the fourth mapping table in write operations.

5 (original): The flash memory access apparatus as claimed in claim 4, wherein the flash memory controller detects errors due to power cutoff during a data write process through the fourth mapping table and during a mapping information update process through the second and third mapping tables, and recovers data related to the errors.

6 (original): A method for accessing a flash memory having a first mapping table containing a physical address of a data block read from the flash memory, comprising the steps of:

receiving a logical address along with a command if read and write operations are requested by a processor;

checking the logical address in a second mapping table containing mapping information, from which error blocks are excluded, of physical address information of the data block read

from the flash memory and a third mapping table containing the most recent mapping information, in order to perform the read and write operations; and

accessing the physical address of a specified data block and performing the read and write operations, when the logical address exists within the second and third mapping tables.

7 (original): The method as claimed in claim 6, further comprising the step of treating the read operation as an error, when the logical address does not exist within the second and third mapping tables.

8 (original): The method as claimed in claim 6, further comprising the step of accessing a physical address of a free block allocated through a fourth mapping table containing free block information created by a flash memory controller and performing the write operation, when the logical address does not exist within the second and third mapping tables.

9 (original): The method as claimed in claim 6, further comprising the step of initializing the flash memory by loading the mapping information for efficiently accessing the flash memory from a map block of the flash memory according to the operations requested by the processor.

10 (original): The method as claimed in claim 6, further comprising the step of detecting errors due to power cutoff occurring during the write operation process and recovering data related to the errors.

11 (original): The method as claimed in claim 6, wherein the read operation comprises the steps of:

receiving a given logical address for a data read, if the read operation is requested by the processor;

translating the given logical address into a physical address of the flash memory into which data is written, by referring to the second and third mapping tables;

determining in the translation step whether the given logical address is a valid address existing within the data block; and

treating the read operation as a read error if the given logical address is not the valid address, or reading data written into the flash memory through the physical address and transmitting the read data to the processor if the given logical address is the valid address.

12 (original): The method as claimed in claim 6, wherein the write operation comprises the steps of:

receiving a given logical address for a data write, if the write operation is requested by the processor;

determining whether the given logical address exists within the second and third mapping tables;

specifying an arbitrary block specified by a stepwise mapping scheme, if the given logical address exists, or searching for a free block allocated through a fourth mapping table containing free block information created by a flash memory controller as a physical address into which data will be written, if the given logical address does not exist;

writing the mapping information between the given logical address and the physical address into the third mapping table according to the specified physical address, and changing a

pointer of the third mapping table indicating a use region according to the updated mapping information;

receiving data transmitted from the processor in a buffer and writing the input data into the physical address;

determining whether errors have occurred during the data write step; and

if errors have occurred according to a result of the determination, writing a block corresponding to the physical address, as an error block, into the first mapping table containing the physical address of the data block read from the flash memory, changing the address information representing current mapping information to an address, into which the updated first mapping table is written, according to updated error information, and searching the free block to attempt to write the data.

13 (original): The method as claimed in claim 12, further comprising the steps of:

allocating a new block for updating the mapping information of the third mapping table into the second mapping table and writing the mapping information of the second mapping table updated by the mapping information of the third mapping table into the allocated block, if a storage capacity of the mapping information of the third mapping table allocated to the flash memory is insufficient due to previous mapping information;

deleting the previous mapping information written into the third mapping table; and

changing pointers of the respective second and third mapping tables indicating the use region, according to the mapping information that is updated or deleted.

14 (original): The method as claimed in claim 12, further comprising the steps of:

allocating a new block for updating the mapping information of the third mapping table into the second mapping table and writing the mapping information of the second mapping table updated by the mapping information of the third mapping table into the allocated block, if storage capacities of the mapping information of the second and third mapping tables allocated to the flash memory are insufficient due to previous mapping information;

deleting the previous mapping information written into the second and third mapping tables; and

changing pointers of the respective second and third mapping tables indicating the use region, according to the mapping information that is updated or deleted.

15 (currently amended): The method as claimed in claim 12, further comprising the steps of:

receiving data input from the processor in the buffer and then merging the input data and previously written data to write the merged data into the physical address, if the previously written data exists within the logical address; and

deleting the data in the block into which the data was previously written, after the data write is completed[[,]].

16 (original): The method as claimed in any one of claims 12 to 14, further comprising the step of storing the previous mapping information in second and third spare blocks of the respective second and third mapping tables, whenever the mapping information of the mapping tables are updated.

17 (original): The method as claimed in claim 9, wherein the step of initializing the flash memory comprises the steps of:

checking entire information on the flash memory containing stored information related to the first mapping table address information written into the specified block of the flash memory;

reading the first mapping table from the map block of the flash memory and reading the second and third mapping tables through respective second and third mapping table regions allocated to the map block;

logically merging the read first, second and third mapping tables to create a fourth mapping table and then storing the created fourth mapping table in a RAM; and

waiting to perform the read or write operations requested by the processor.

18 (original): The method as claimed in claim 17, wherein the step of reading the first mapping table comprises the steps of:

searching for the address information of the first mapping table from the specified block by using the stored information;

detecting the address information and searching the first mapping table from the first mapping table block region allocated to the flash memory so as to write the address information searched in the first mapping table into the specified block, if the address information exists, or searching the first mapping table block region so as to generate the first mapping table and then to write the address information generated in the first mapping table into the specified block, if the address information does not exists; and

reading out the first mapping table from the flash memory by using the searched or generated address information.

19 (original): The method as claimed in claim 10, wherein the step of detecting the errors and recovering the relevant data comprises the steps of:

checking the second and third mapping tables from the map block of the flash memory to determine whether a plurality of the second and third mapping tables currently used exist within the map block;

if the plurality of second and third mapping tables exist in the map block, determining that the errors have occurred due to power cutoff during the process of updating the mapping information and deleting the most recent mapping information written into the second and third mapping tables;

merging the checked second and third mapping tables or the second and third mapping table with the most recent mapping information deleted therefrom and the first mapping table read out from the flash memory, into the fourth mapping table generated by the flash memory controller during an initialization process of the flash memory;

searching a first free block form the merged fourth mapping table and determining whether the searched free block is a pure free block into which no data are written; and

if the searched free block is not the pure free block, determining that the errors have occurred due to power cutoff during the process of writing the data and deleting the written data.

20 (new): The flash memory as claimed in claim 1, wherein the first mapping table includes physical block numbers for corresponding plurality of blocks and does not include logical block numbers, the second mapping table includes mapping information between the

logical block numbers and the physical block numbers and the third mapping table includes at least one of the logical block numbers and at least one updated physical block number.

21 (new): The flash memory as claimed in claim 1, wherein the first mapping table, the second mapping table, and the third mapping table are logically summed to obtain a fourth mapping table.

22 (new): The flash memory as claimed in claim 21, wherein the fourth mapping table indicates free blocks and does not indicate used blocks.